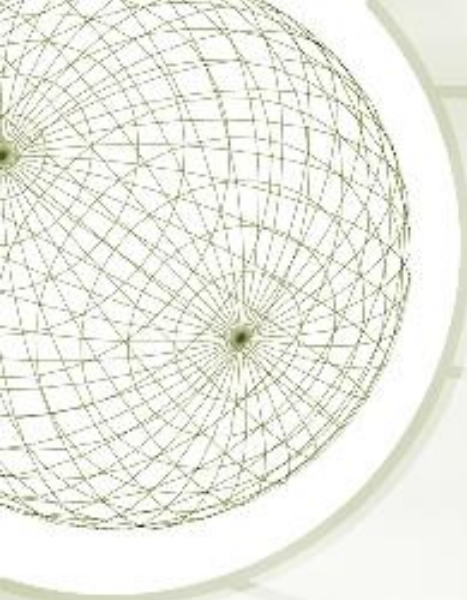
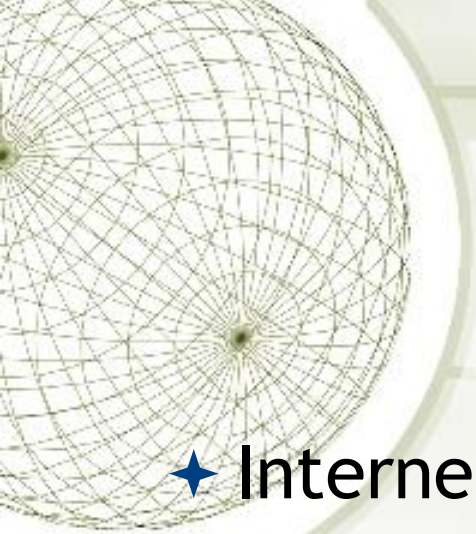


IP Security

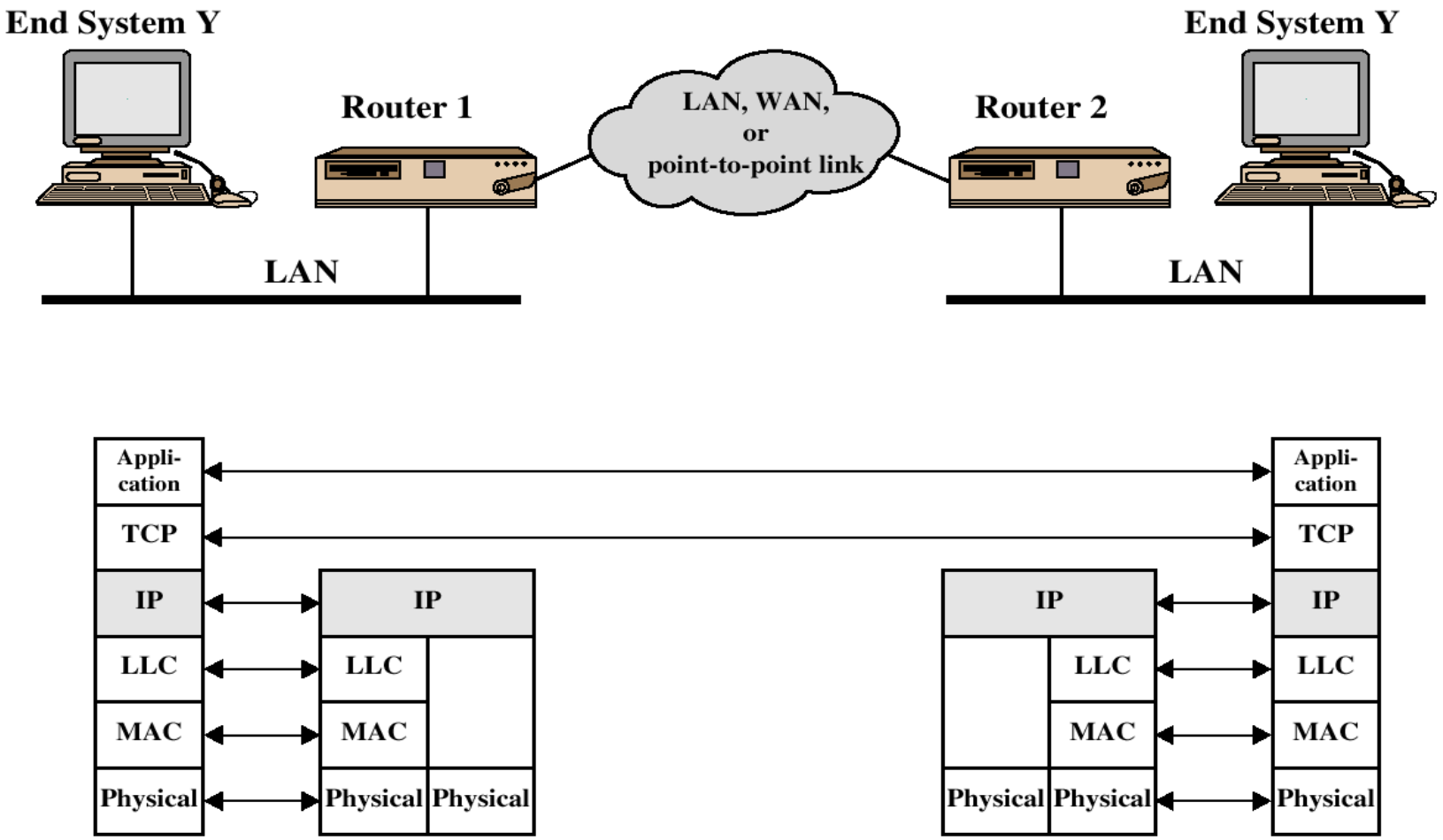
Ola Flygt
Linnaeus University, Sweden
http://homepage.lnu.se/staff/oflmsi/
Ola.Flygt@lnu.se
+46 470 70 86 49



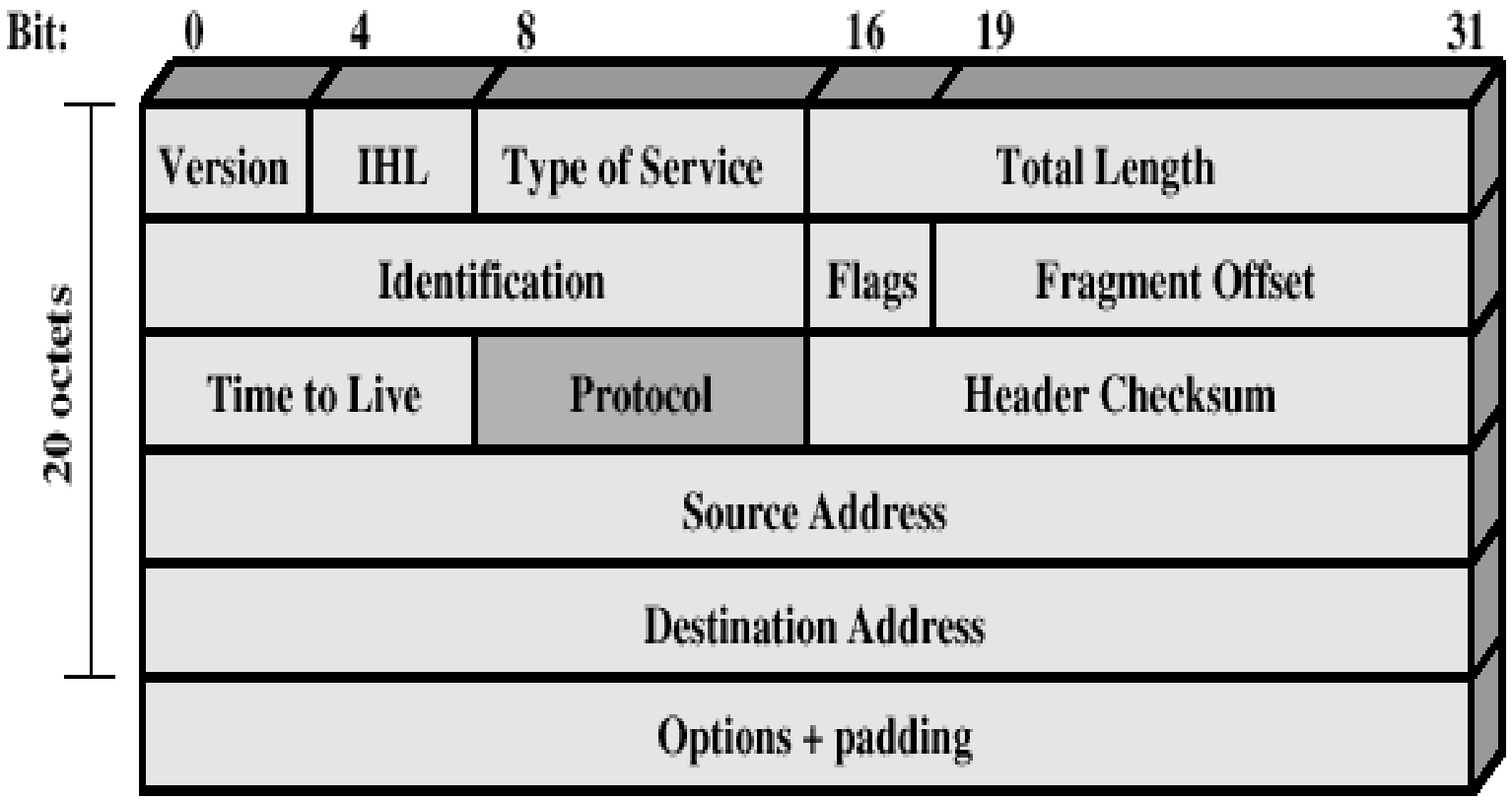
Outline

- Internetworking and Internet Protocols
- IP Security Overview
- IP Security Architecture
- Authentication Header
- Encapsulating Security Payload
- Combinations of Security Associations
- Key Management

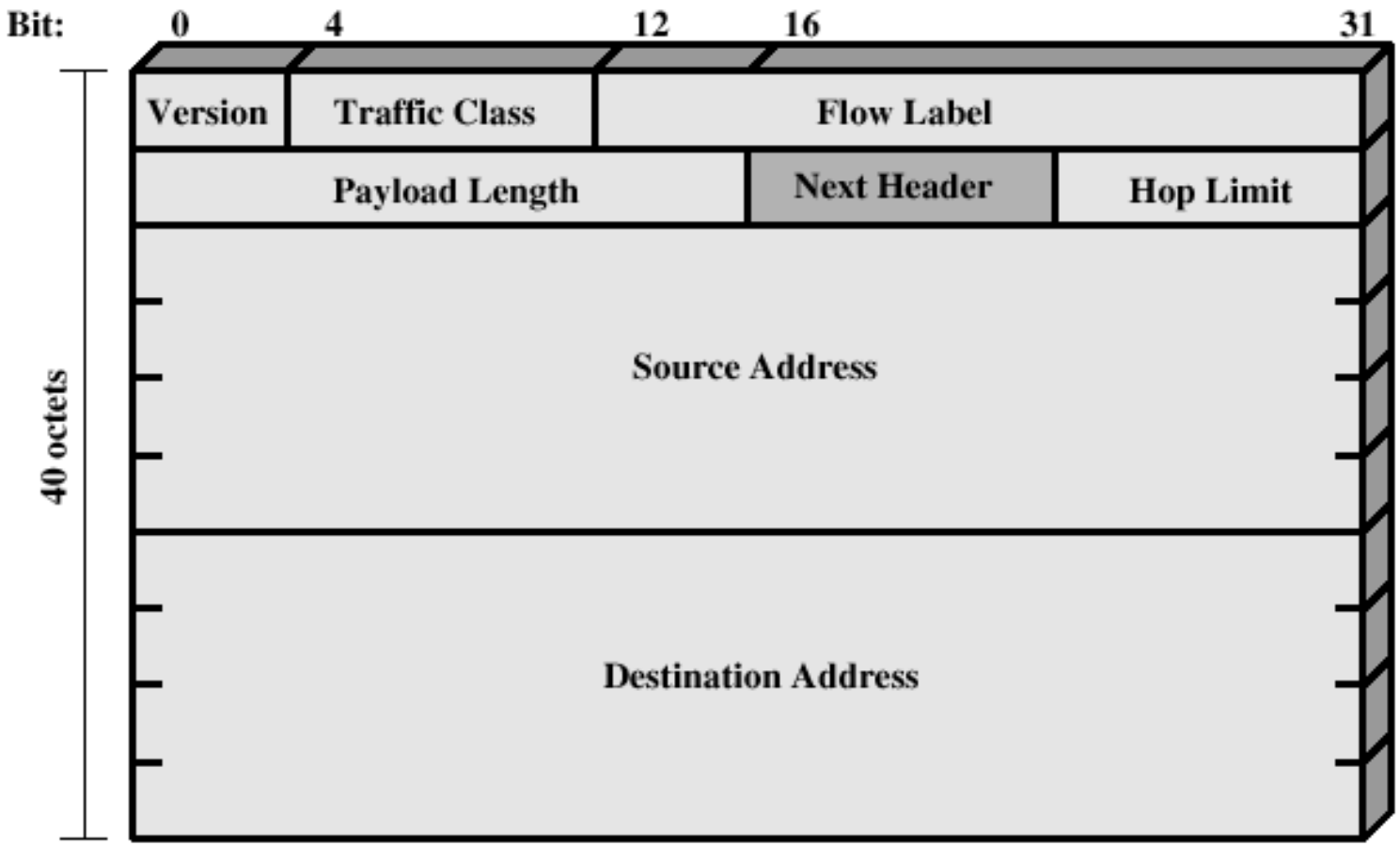
TCP/IP Example



IPv4 Header


Bit:
0
4
8
16
19
31
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options + padding
20 octets

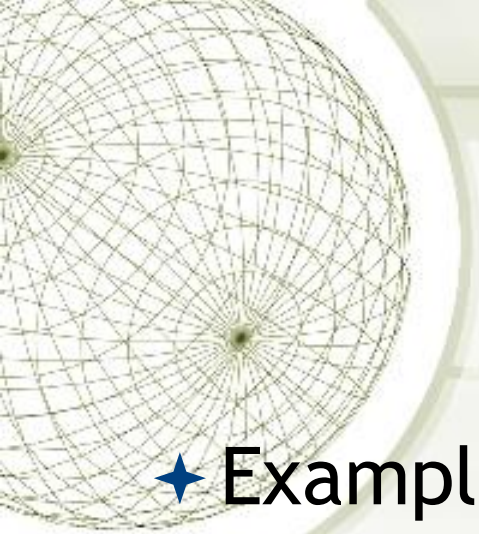
IPv6 Header


Bit:
0
4
12
16
31
Version
Traffic Class
Flow Label
Payload Length
Next Header
Hop Limit
Source Address
Destination Address
40 octets

IP Security Overview

- IPsec is not a single protocol

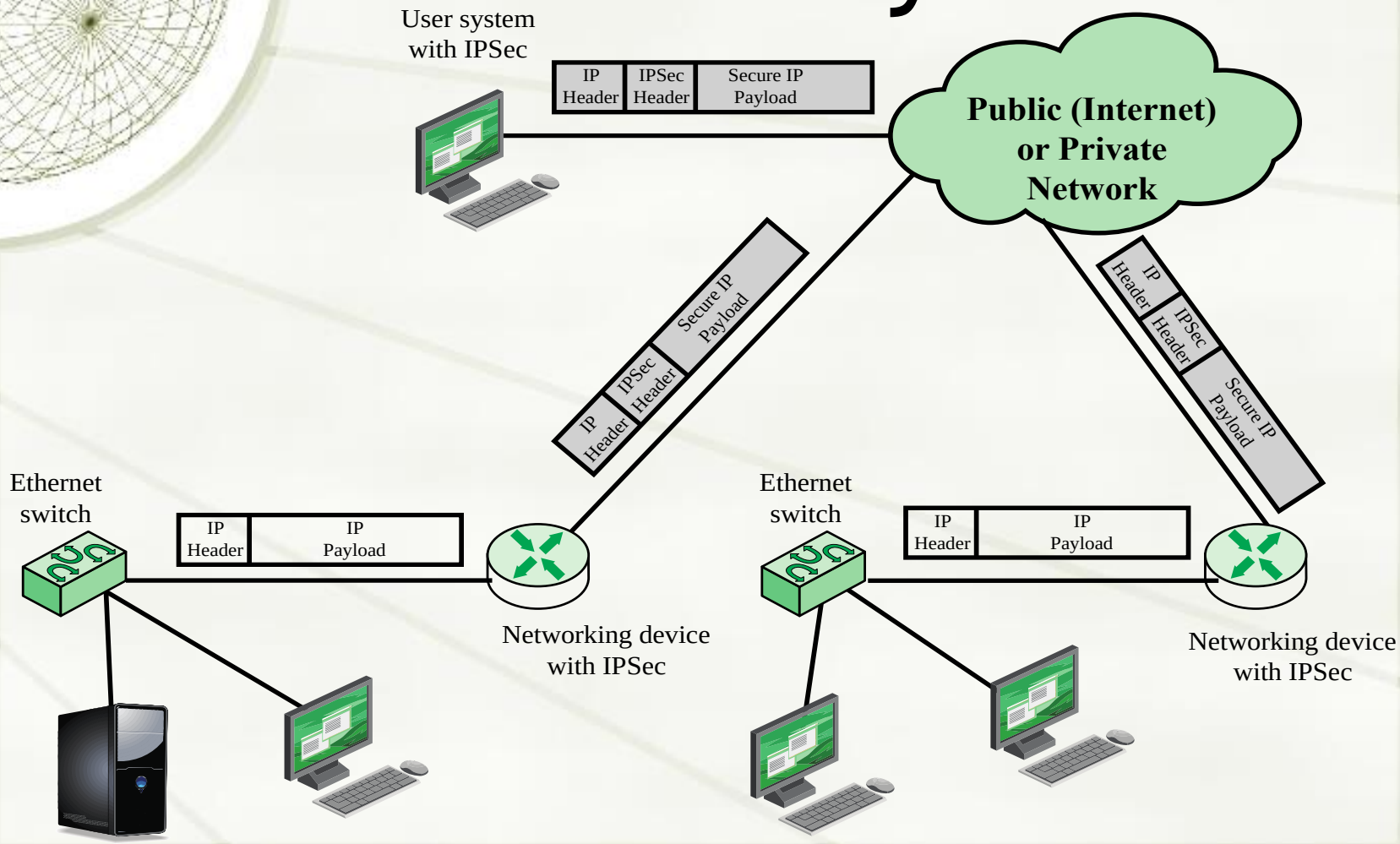
Instead, IPsec provides a set of standards, security algorithms plus a general framework that allows a pair of communicating entities to use whichever algorithms they decide will provide the security appropriate for the communication.



IP Security Overview

- Examples of applications of IPsec
  - Secure branch office connectivity over the Internet
  - Secure remote access over the Internet
  - Establishing extranet and intranet connectivity with partners
  - Enhancing electronic commerce security
- A requirement is that both sides of an IPsec connection are managed (unlike TLS)

IP Security Scenarios



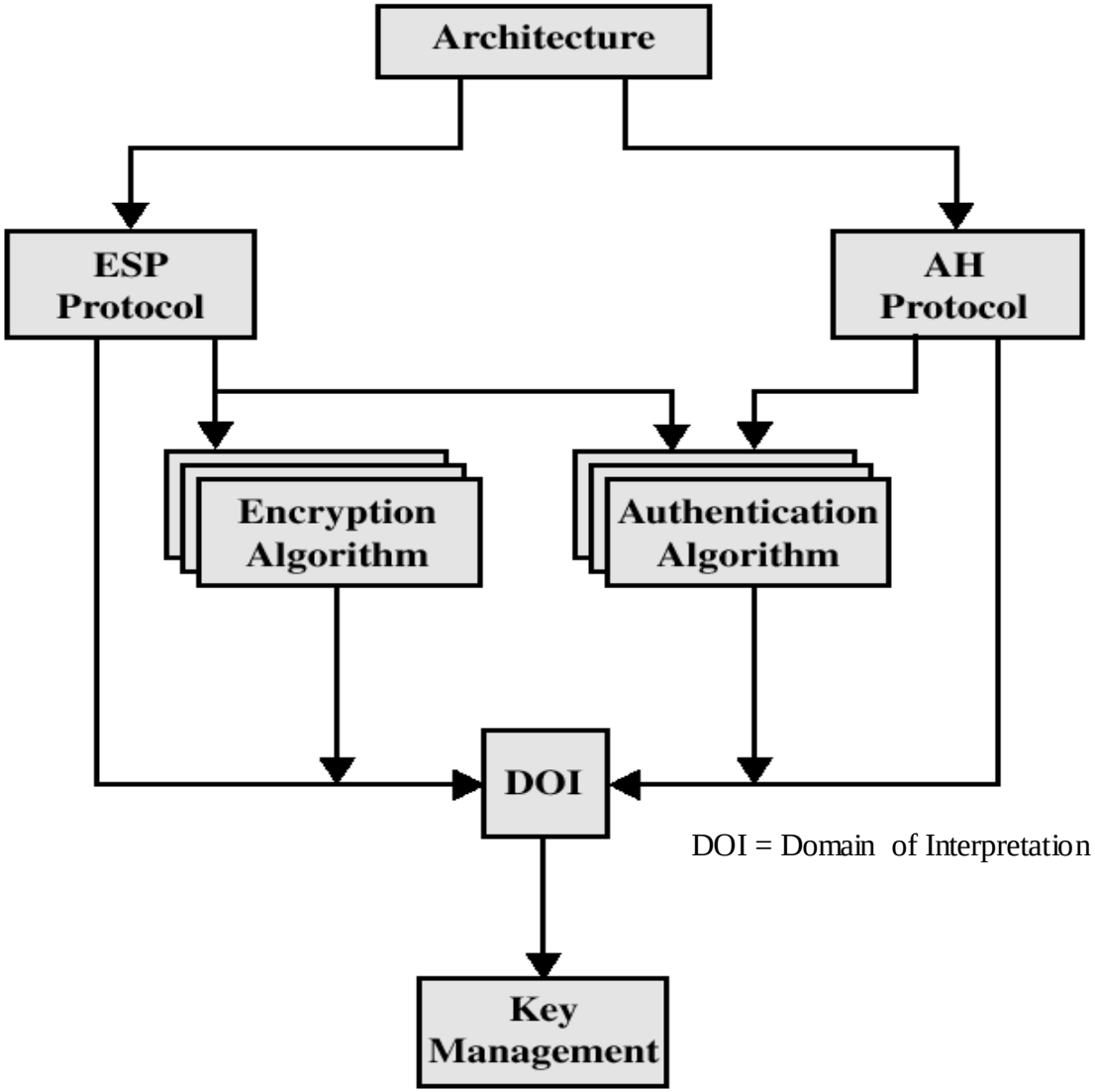
IP Security Overview

- Benefits of IPsec
  - Transparent to applications (it is below the transport layer (TCP, UDP)
  - Provide security for sites and individual users
- Additionally, IPsec can assure that:
  - A router or neighbour advertisement comes from an authorized router
  - A redirect message comes from the router to which the initial packet was sent
  - A routing update is not forged

IP Security Architecture

- Specification is quite complex, with groups:
  - Architecture
    - RFC 4301 *Security Architecture for Internet Protocol*
  - Authentication Header (AH)
    - RFC 4302 *IP Authentication Header*
  - Encapsulating Security Payload (ESP)
    - RFC 4303 *IP Encapsulating Security Payload (ESP)*
  - Internet Key Exchange (IKE)
    - RFC 7296 *Internet Key Exchange (IKEv2) Protocol*
  - Cryptographic algorithms
  - Others, including those dealing with security policy and management information base (MIB) content

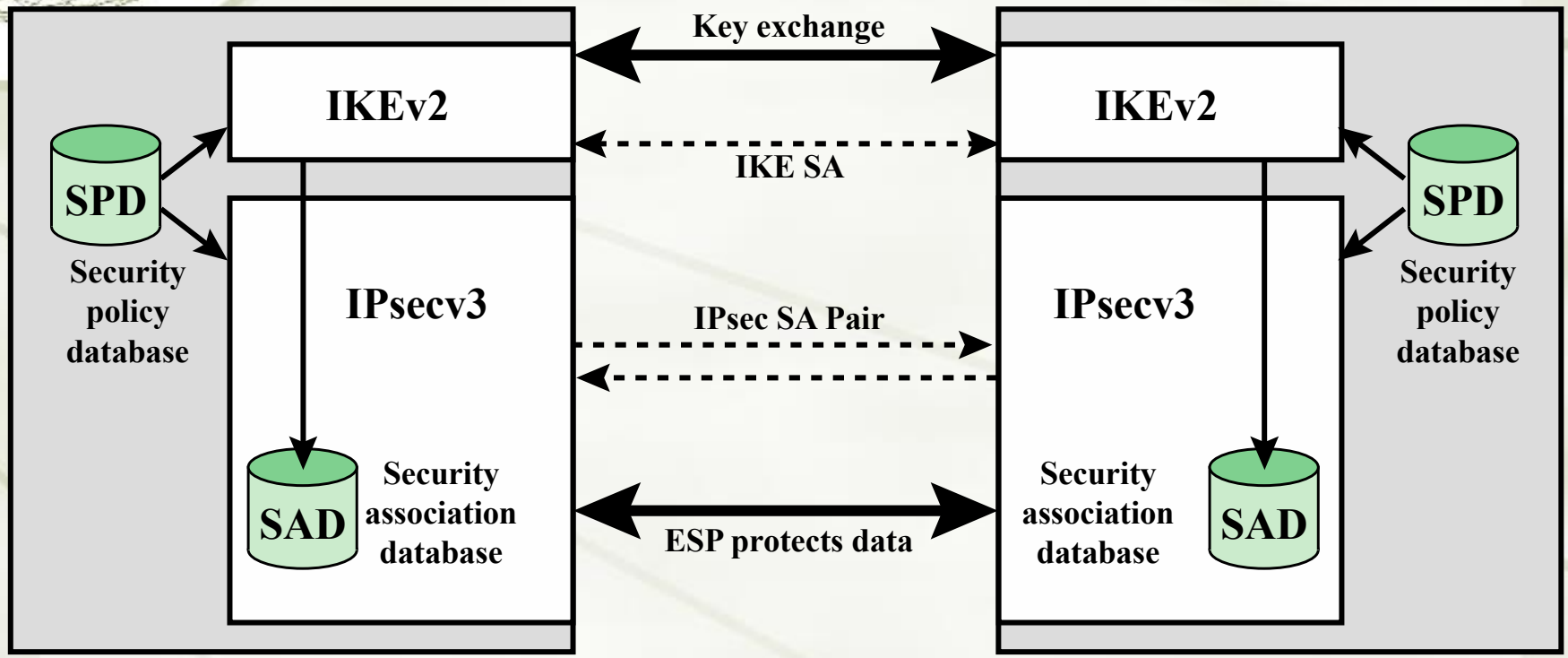
IPsec Document Overview



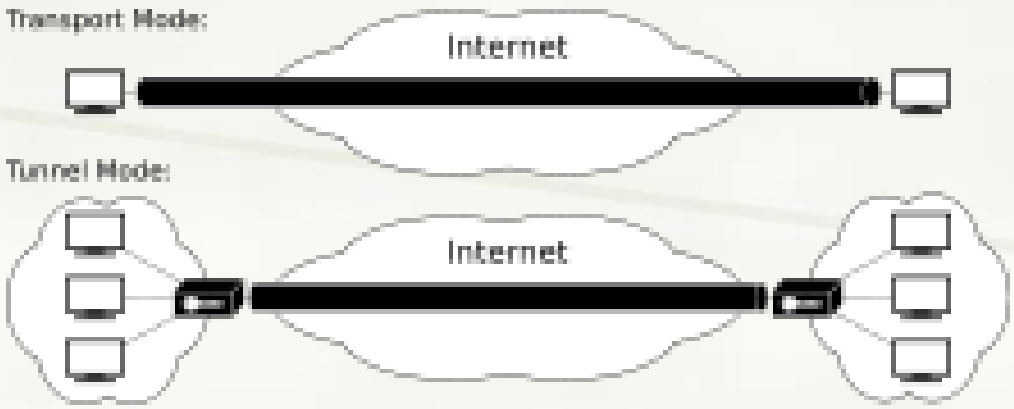
IPsec Services

- Access Control
- Message integrity (Connectionless integrity)
- Data origin authentication
- Rejection of replayed packets (a form of partial sequence integrity)
- Confidentiality (encryption)
- Limited traffic flow confidentiality

IPsec Architecture



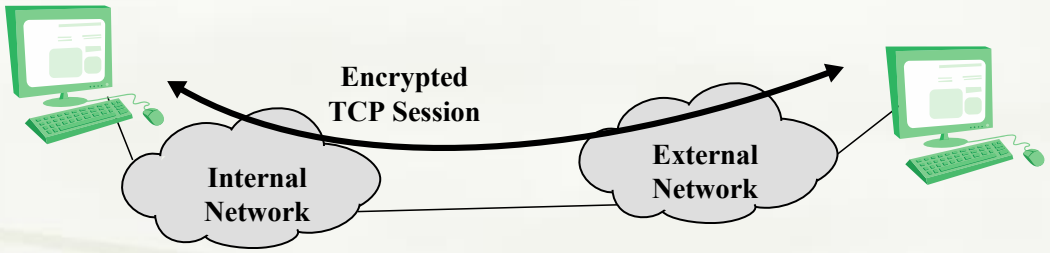
Transport and Tunnel Modes



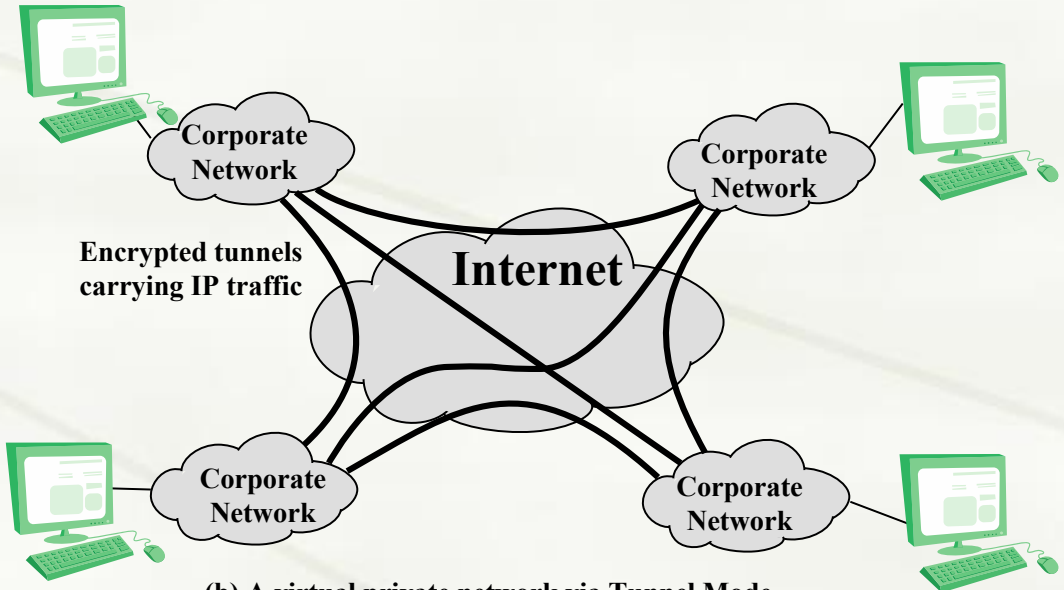
- Transport Mode
  - to encrypt & optionally authenticate IP data
  - can do traffic analysis but is less efficient
  - good for ESP host to host traffic
- Tunnel Mode
  - encrypts entire IP packet
  - add new header for next hop
  - no routers on way can examine inner IP header
  - good for VPNs, gateway to gateway security

	Transport Mode SA	Tunnel Mode SA
AH	Authenticates IP payload and selected portions of IP header and IPv6 extension headers	Authenticates entire inner IP packet plus selected portions of outer IP header
ESP	Encrypts IP payload and any IPv6 extension header	Encrypts inner IP packet
ESP with authentication	Encrypts IP payload and any IPv6 extension header. Authenticates IP payload but no IP header	Encrypts inner IP packet. Authenticates inner IP packet.

Transport and Tunnel Modes



(a) Transport-level security



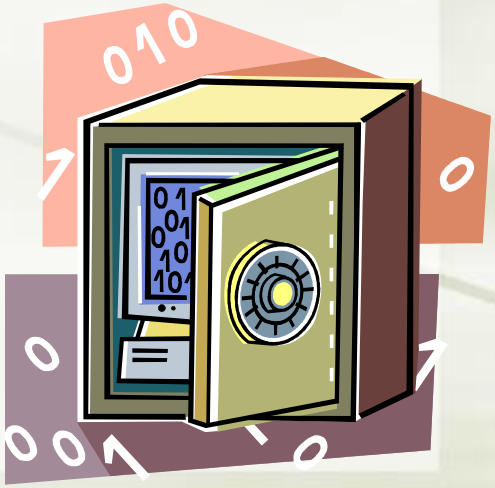
(b) A virtual private network via Tunnel Mode

Security Associations (SA)

- a one-way relationship between sender & receiver that affords security for traffic flow
- defined by 3 parameters:
  - Security Parameters Index (SPI)
  - IP Destination Address
  - Security Protocol Identifier
- Each IPsec node have a database of Security Associations

Security Association Database (SAD)

- Defines the parameters associated with each SA
- Normally defined by the following parameters in a SAD entry:
  - Security parameter index
  - Sequence number counter
  - Sequence counter overflow
  - Anti-replay window
  - AH information
  - ESP information
  - Lifetime of this security association
  - IPsec protocol mode
  - Path MTU

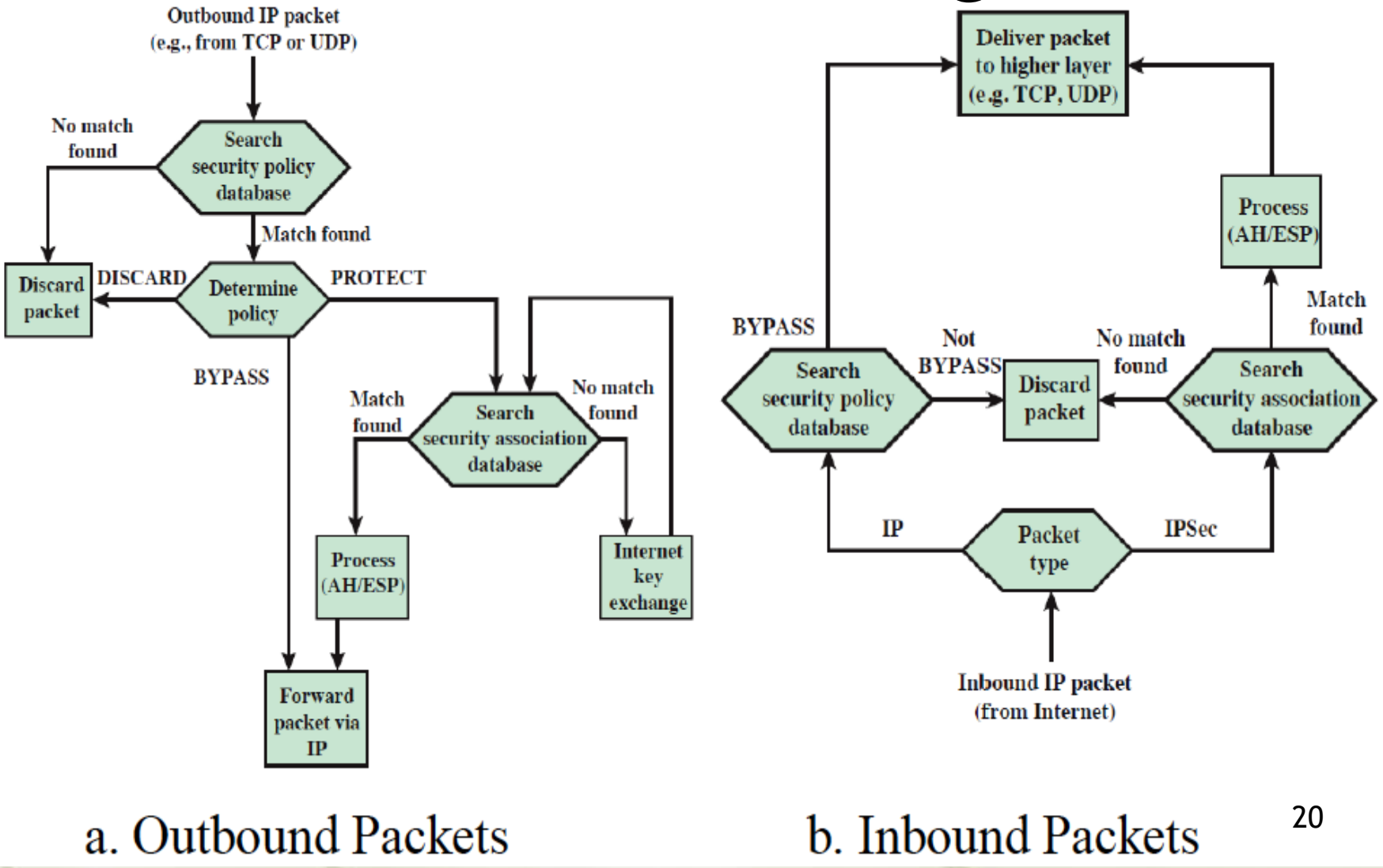


Security Policy Database

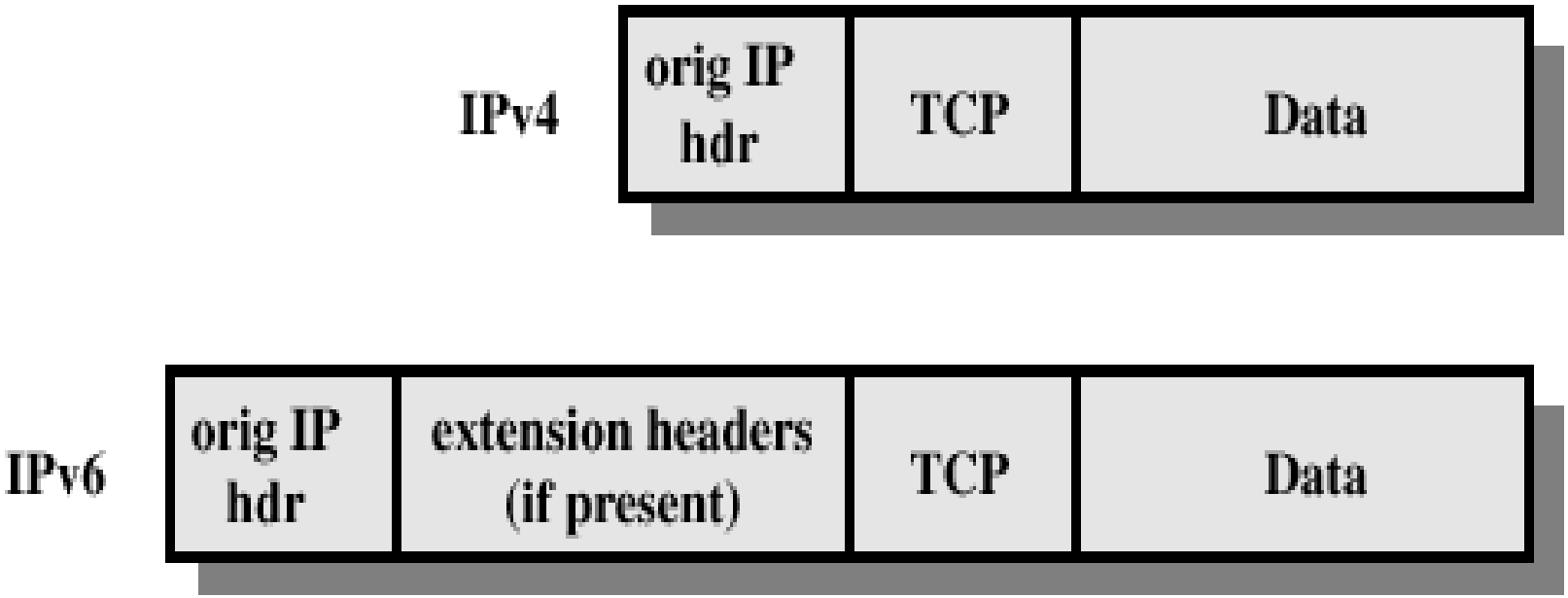
- relates IP traffic to specific SAs
  - match subset of IP traffic to relevant SA
  - use selectors to filter outgoing traffic to map
  - based on: local & remote IP addresses, next layer protocol, name, local & remote ports

Protocol	Local IP	Port	Remote IP	Port	Action	Comment
UDP	1.2.3.101	500	*	500	BYPASS	IKE
ICMP	1.2.3.101	*	*	*	BYPASS	Error messages
*	1.2.3.101	*	1.2.3.0/24	*	PROTECT: ESP intransport-mode	Encrypt intranet traffic
TCP	1.2.3.101	*	1.2.4.10	80	PROTECT: ESP intransport-mode	Encrypt to server
TCP	1.2.3.101	*	1.2.4.10	443	BYPASS	TLS: avoid double encryption
*	1.2.3.101	*	1.2.4.0/24	*	DISCARD	Others in DMZ
*	1.2.3.101	*	*	*	BYPASS	Internet

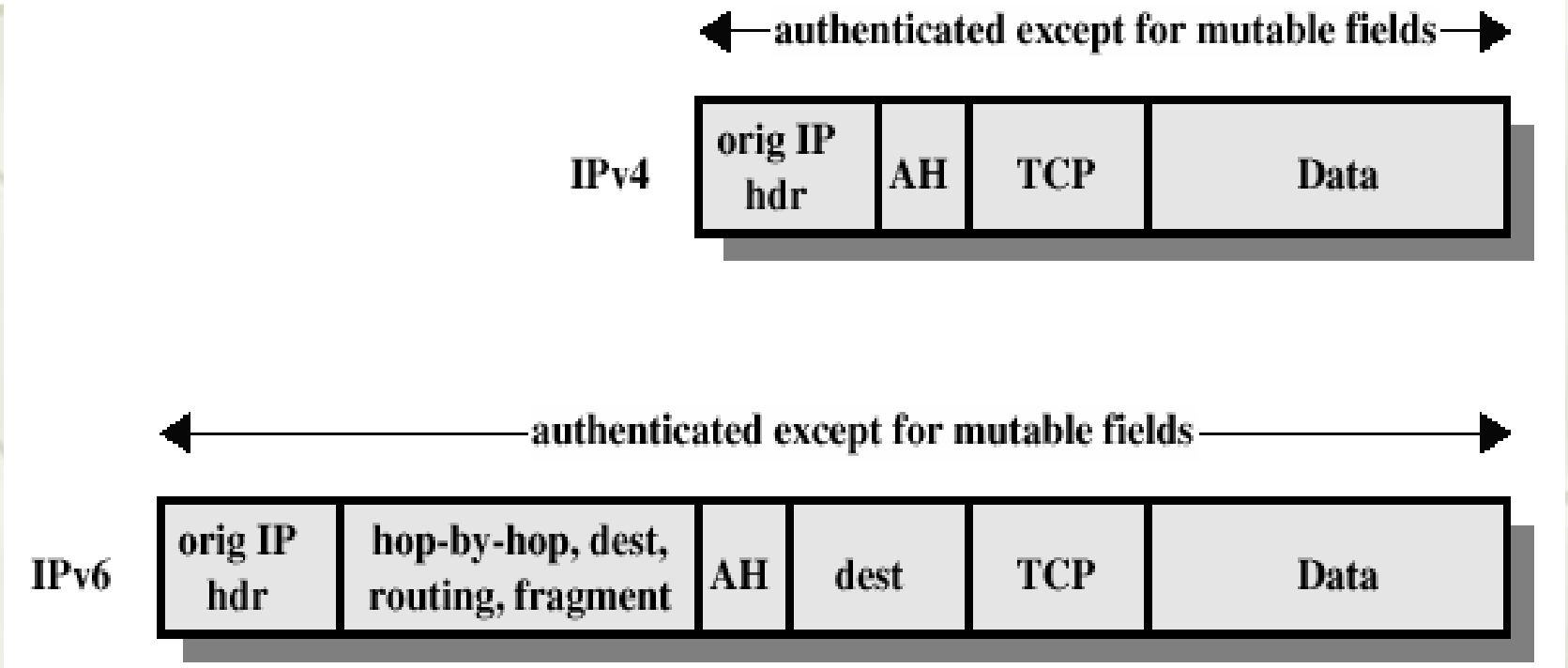
Processing models



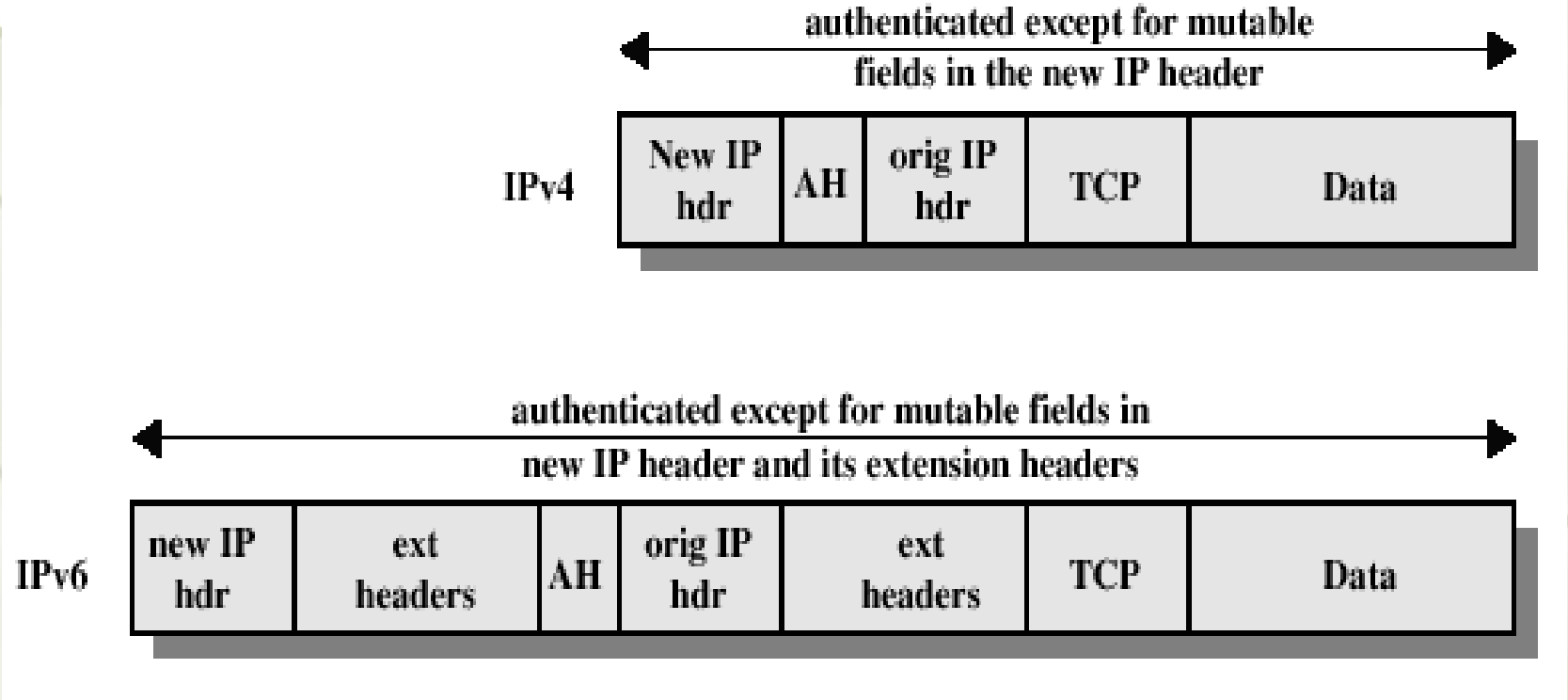
Before applying AH



Transport Mode (AH Authentication)

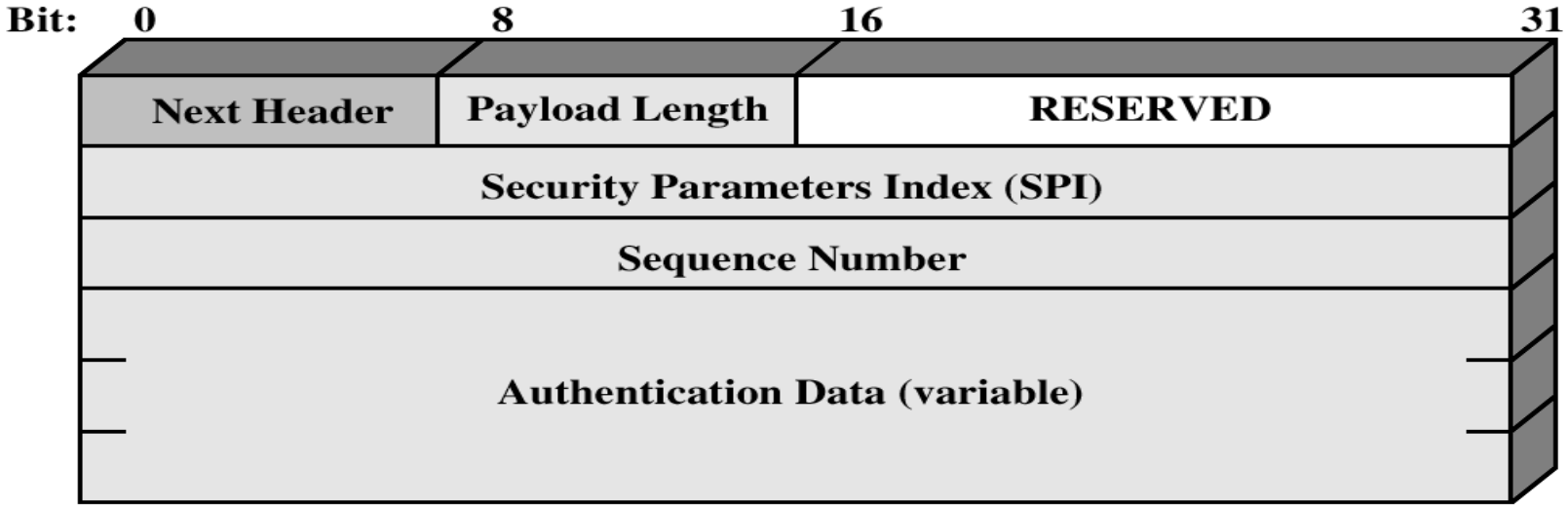


Tunnel Mode (AH Authentication)

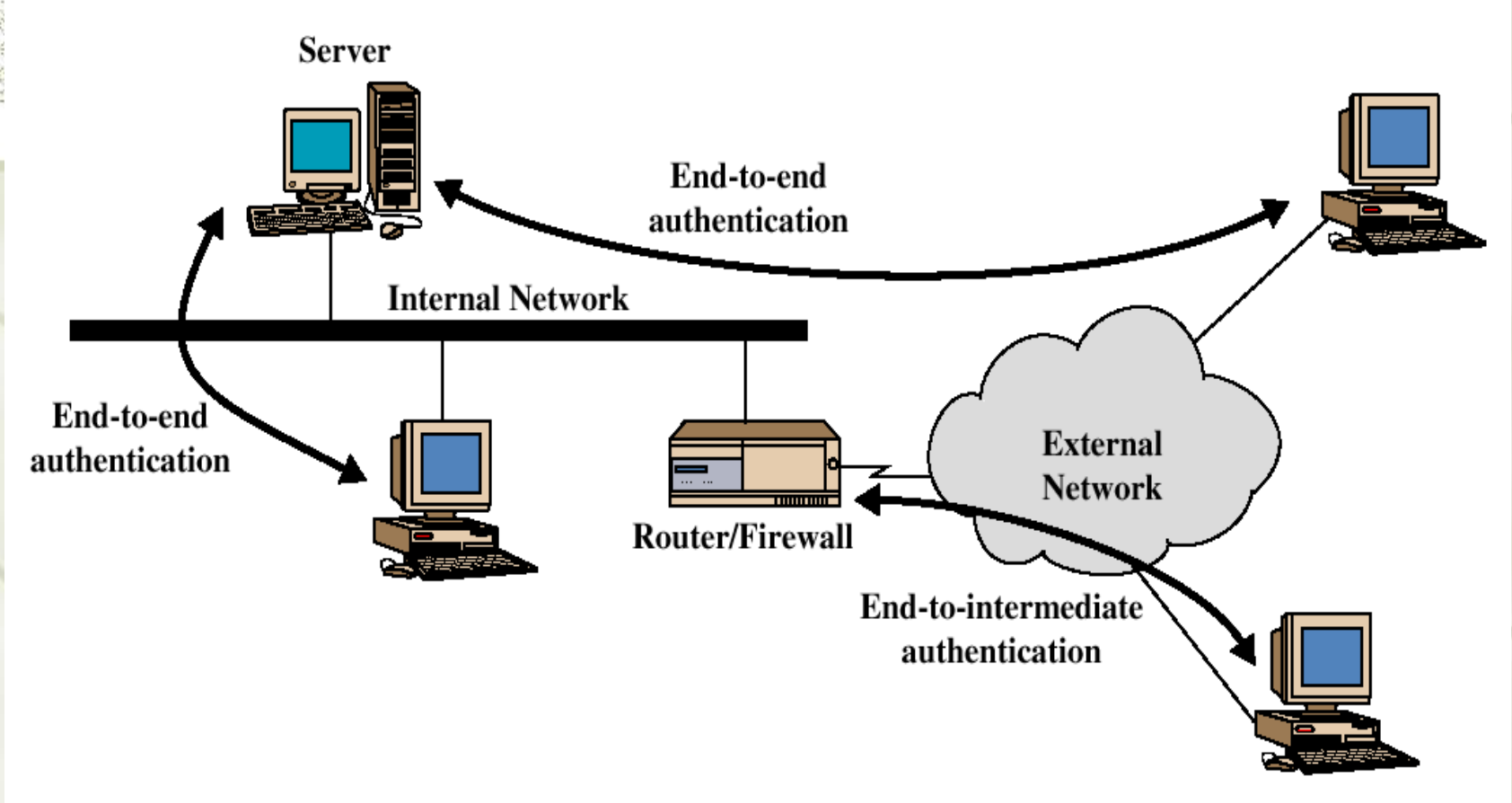


Authentication Header

- Provides support for data integrity and authentication (MAC code) of IP packets.
- Guards against replay attacks.



End-to-end versus End-to-Intermediate Authentication

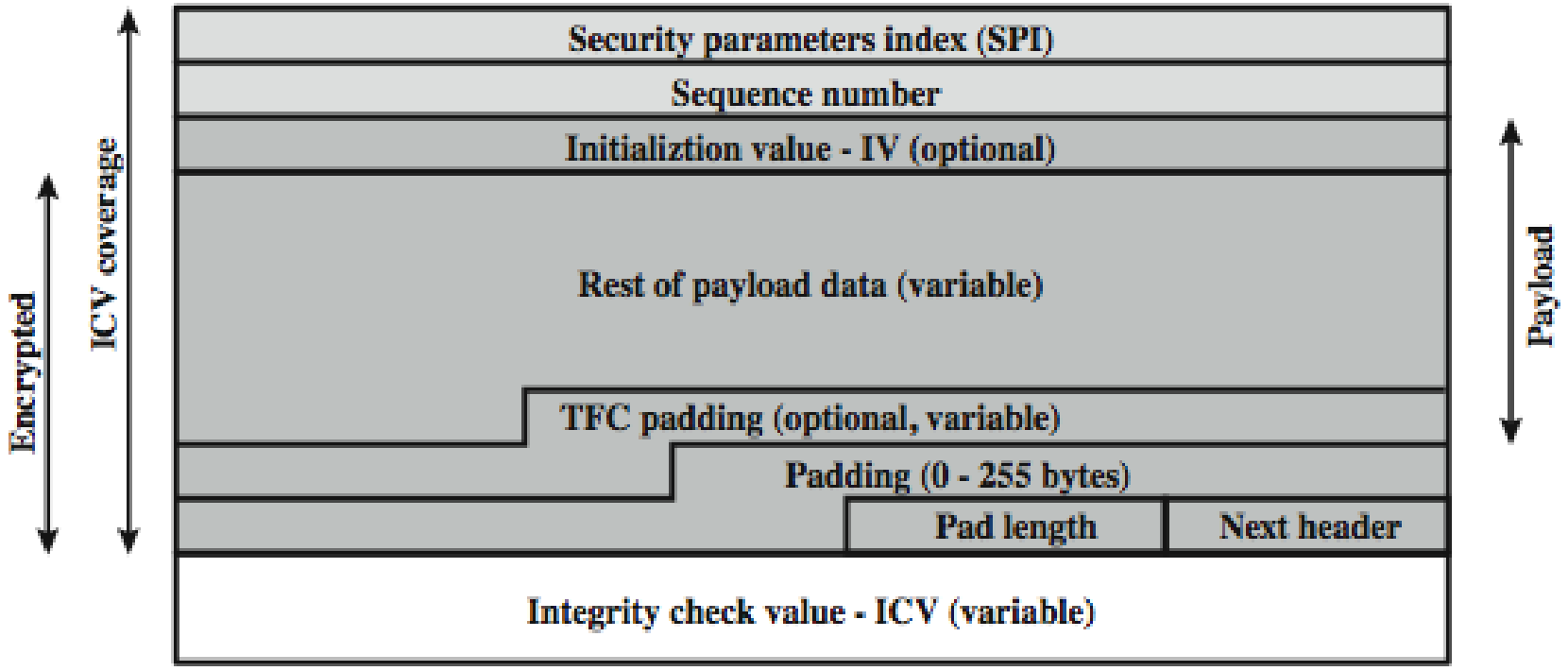


Encapsulating Security Payload (ESP)

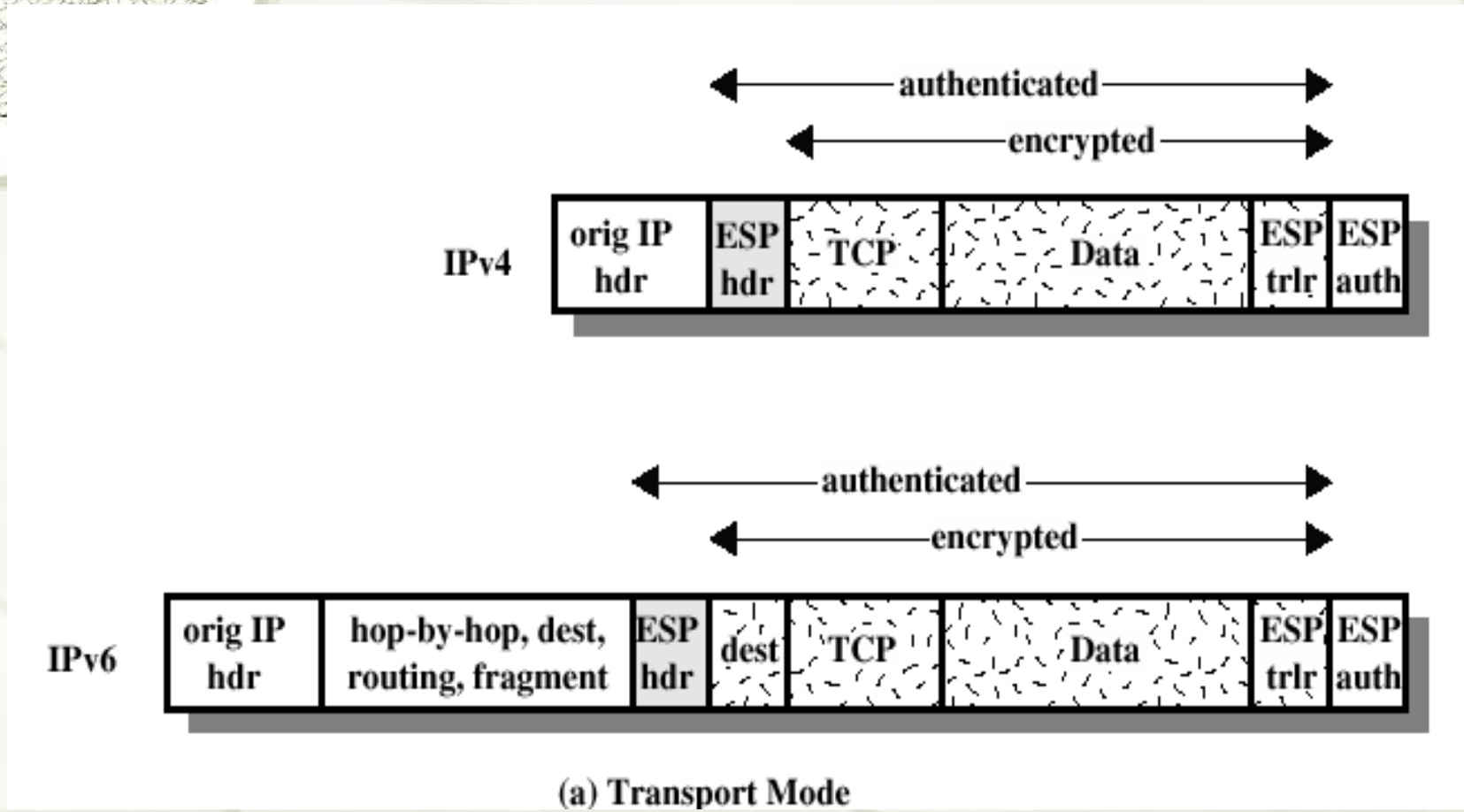
- provides message content confidentiality, data origin authentication, connectionless integrity, an anti-replay service, limited traffic flow confidentiality
- services depend on options selected when establish Security Association (SA), net location
- can use a variety of encryption & authentication algorithms

Encapsulating Security Payload

- ESP provides confidentiality services

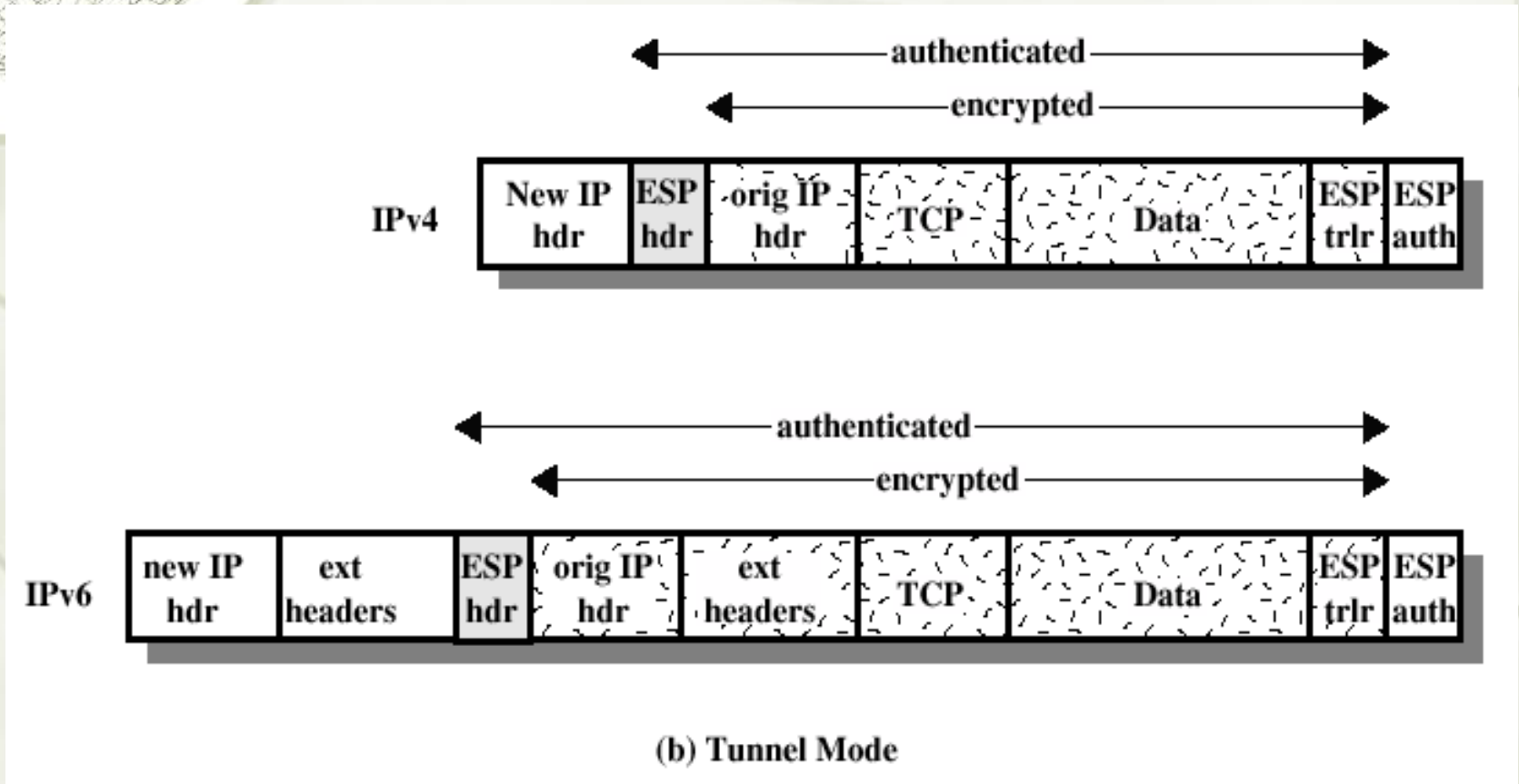


ESP Encryption and Authentication



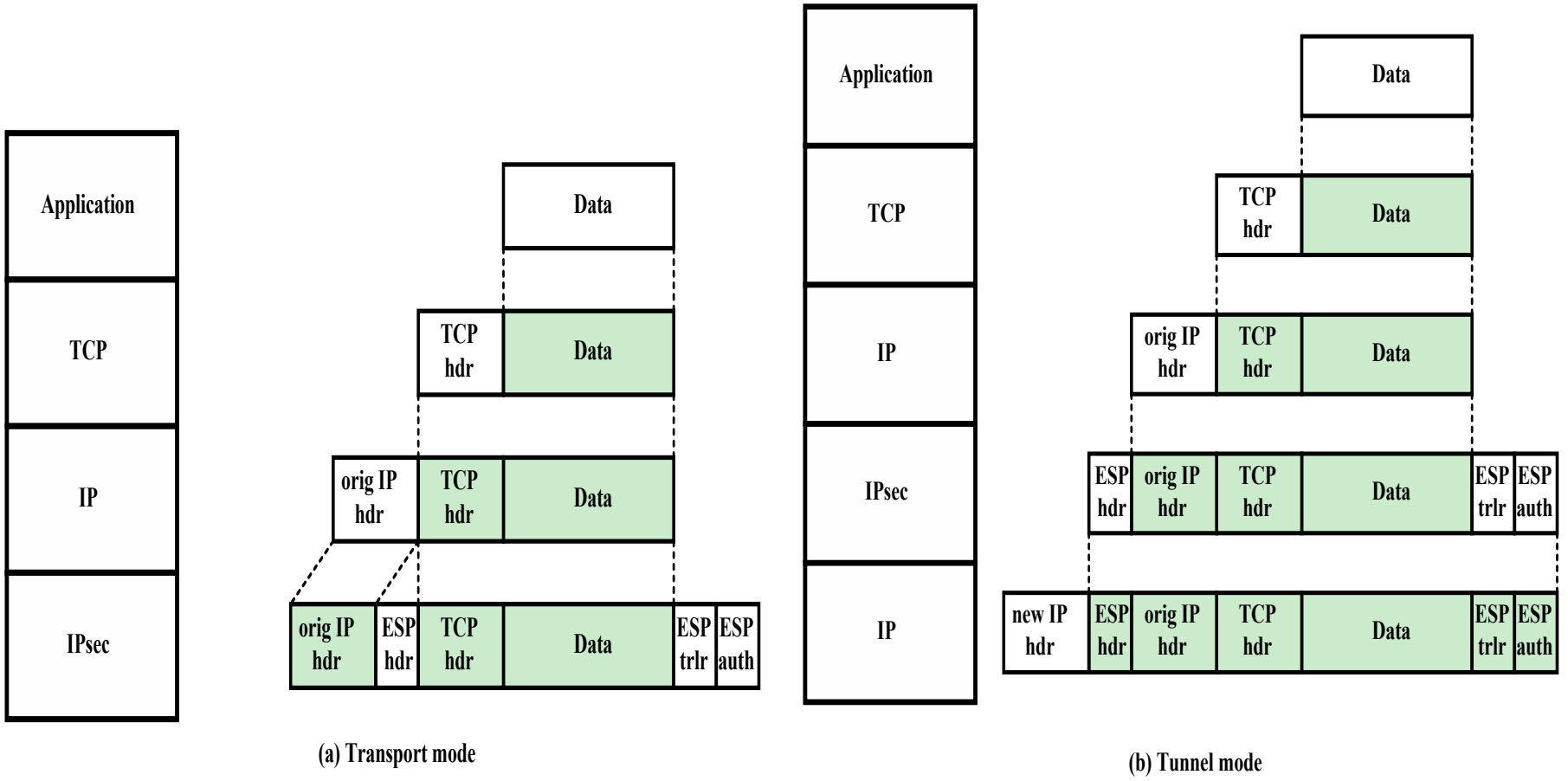
(a) Transport Mode

ESP Encryption and Authentication

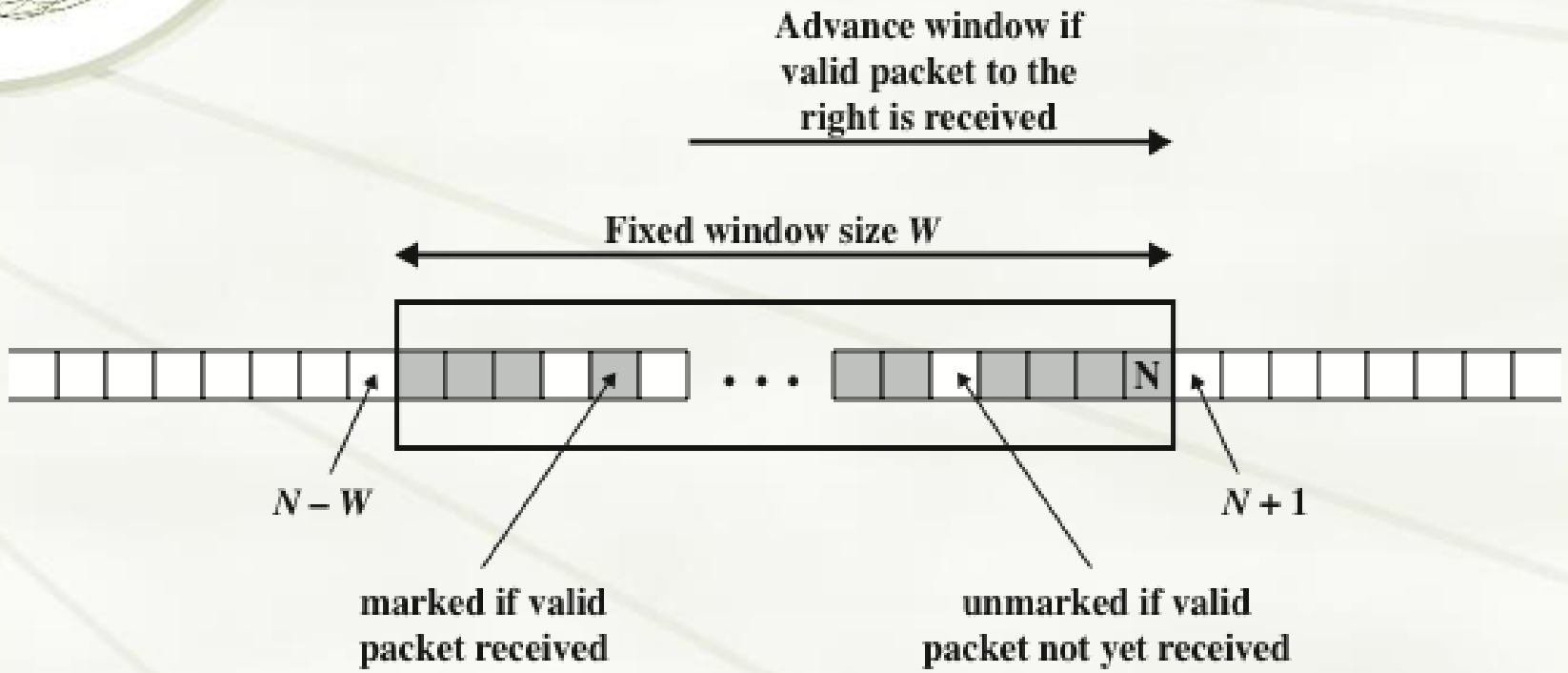


(b) Tunnel Mode

Protocol Operation for ESP



Anti-Replay Mechanism



Combining Security Associations

- An individual SA can implement either the AH or ESP protocol but not both
- *Security association bundle*
  - Refers to a sequence of SAs through which traffic must be processed to provide a desired set of IPsec services
  - The SAs in a bundle may terminate at different endpoints or at the same endpoint
- May be combined into bundles in two ways:

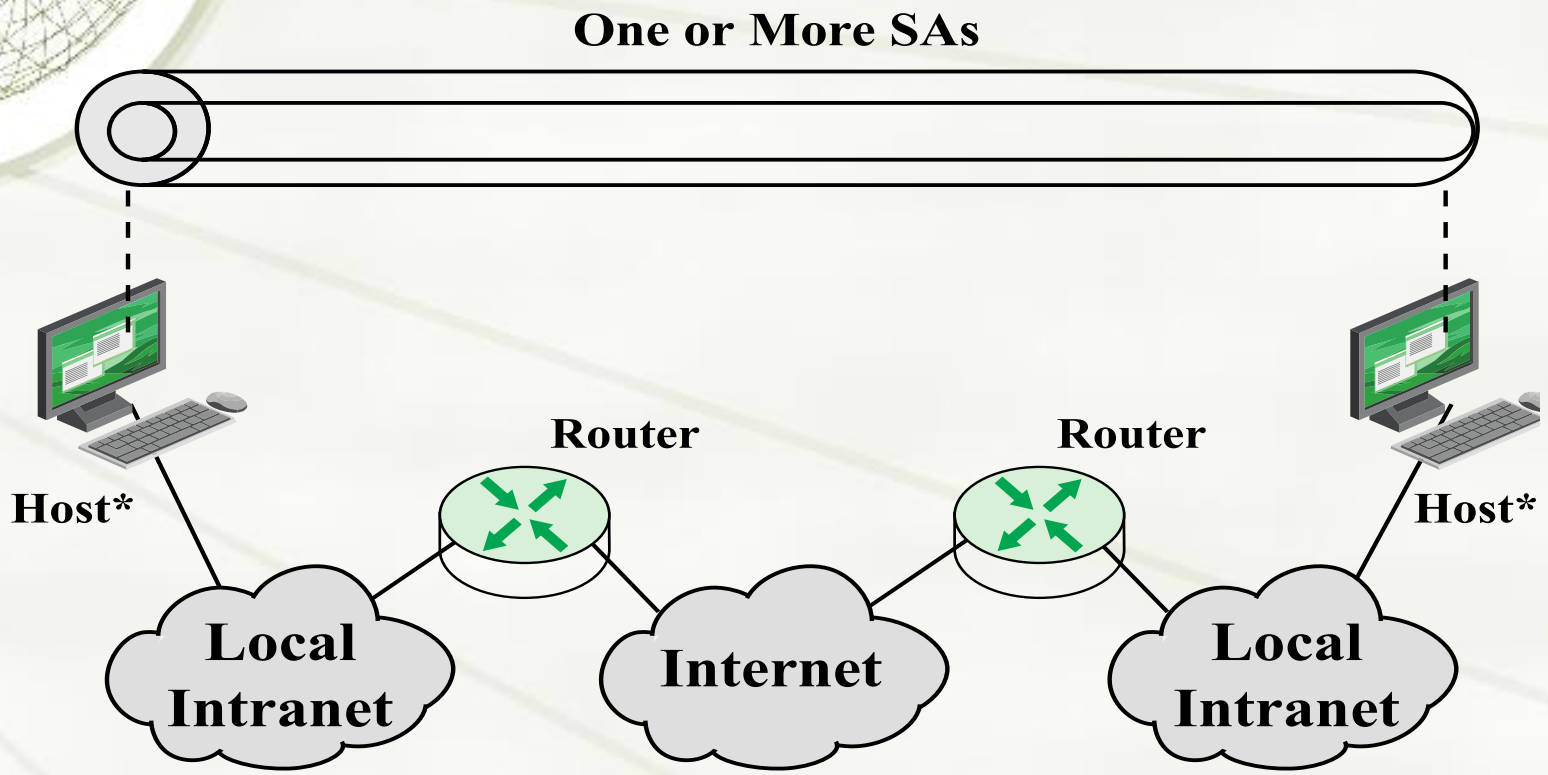
Transport adjacency

- Refers to applying more than one security protocol to the same IP packet without invoking tunneling
- This approach allows for only one level of combination

Iterated tunneling

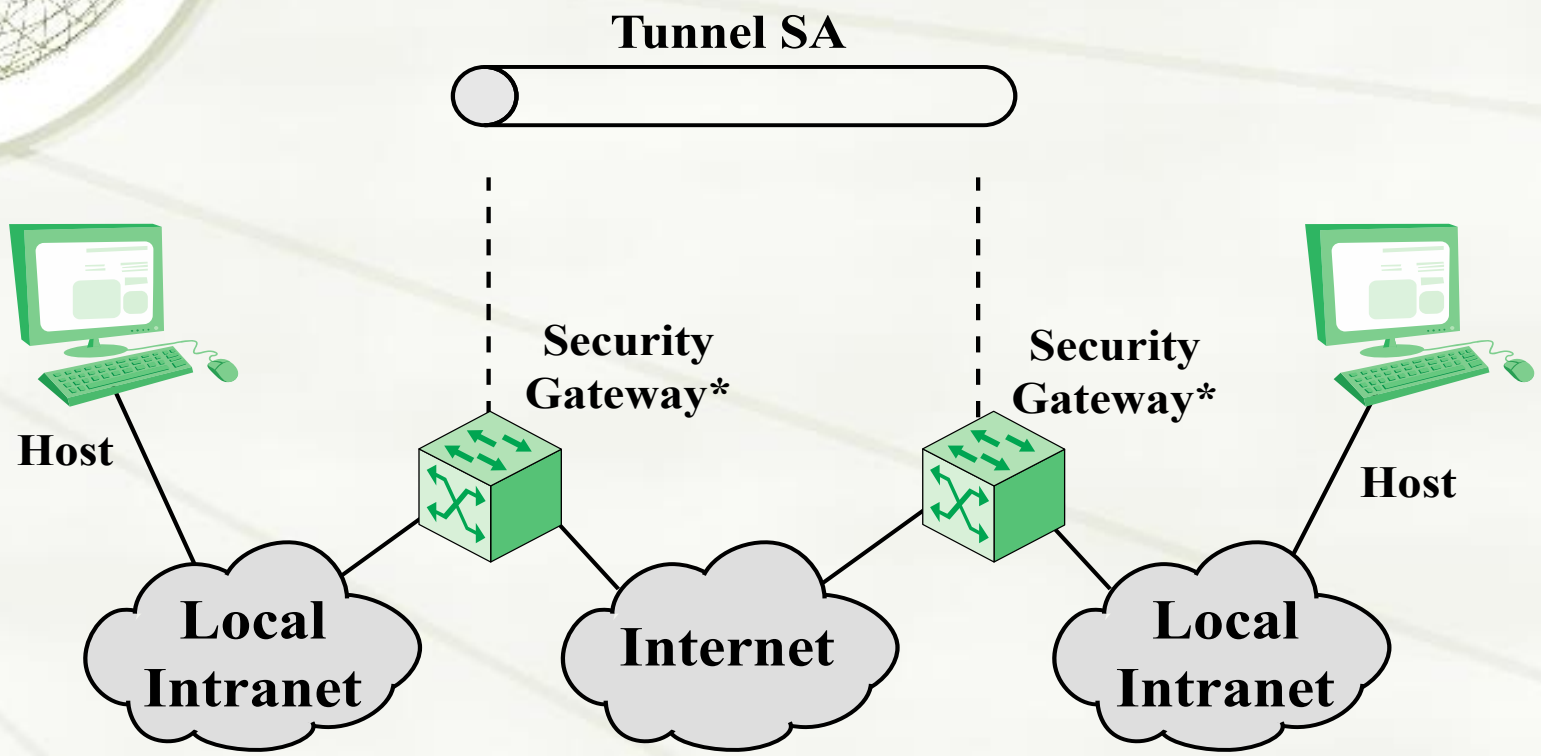
- Refers to the application of multiple layers of security protocols effected through IP tunneling
- This approach allows for multiple levels of nesting

Combinations of Security Associations



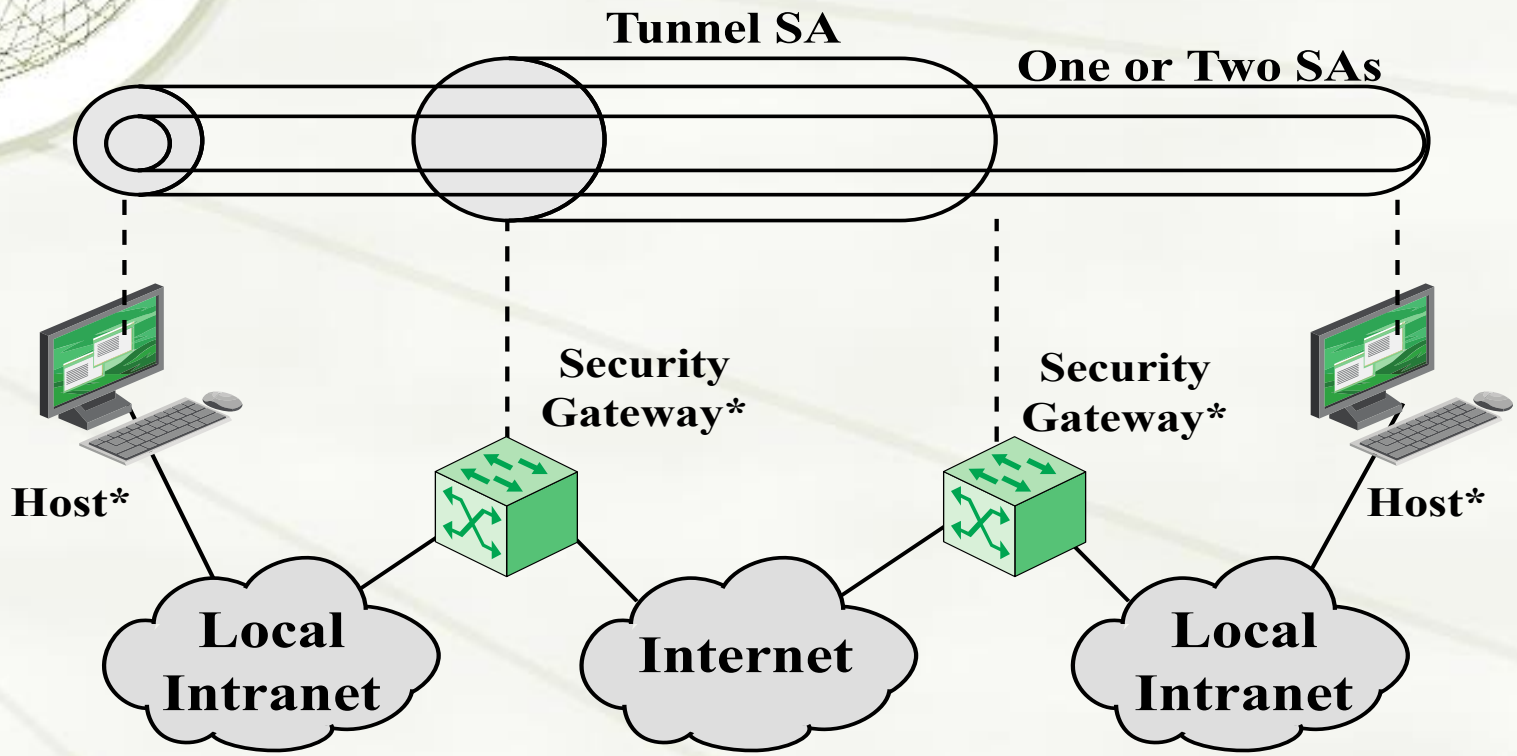
(a) Case 1

Combinations of Security Associations



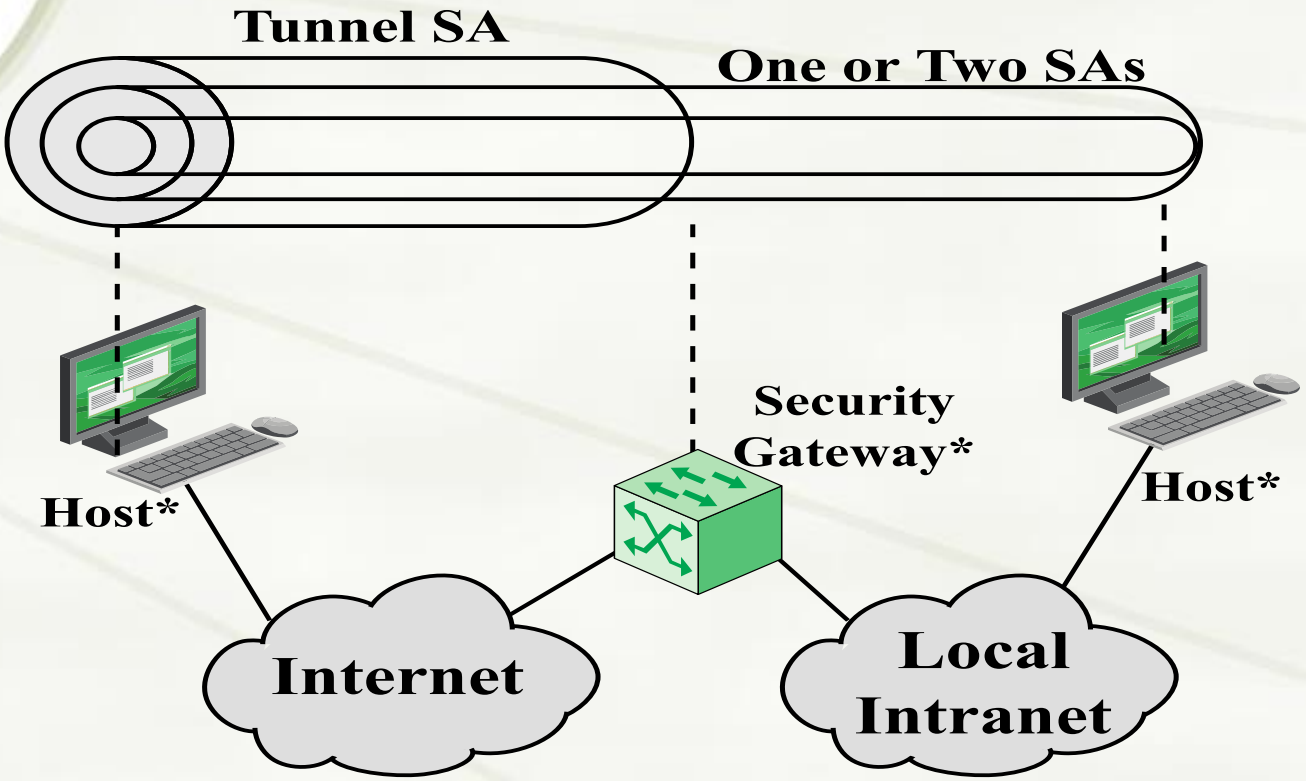
(b) Case 2

Combinations of Security Associations



(c) Case 3

Combinations of Security Associations



(d) Case 4

IPsec Key Management

- handles key generation & distribution
- typically need 2 pairs of keys
  - 2 per direction for AH & ESP
- two options:
  - manual key management
    - sysadmin manually configures every system
  - automated key management
    - automated system for on demand creation of keys for SA's in large systems
    - has Oakley & ISAKMP elements

Internet Key Exchange (IKE)

- IKE=ISAKMP+Oakley
  - automated system for on-demand creation and distribution of keys for enabling SA's in large systems in a protected manner
- Typically SAs need 2 pairs of keys
  - 2 per direction for AH & ESP
- Perfect forward secrecy desired ➔ D-H

Oakley

- A key exchange protocol based on Diffie-Hellman key exchange
- Adds features to address weaknesses
  - Cookies (thwart clogging attacks)
  - groups (global parameters)
  - nonces (against replay attacks)
  - DH key exchange with authentication (thwart MITM attacks)

Oakley

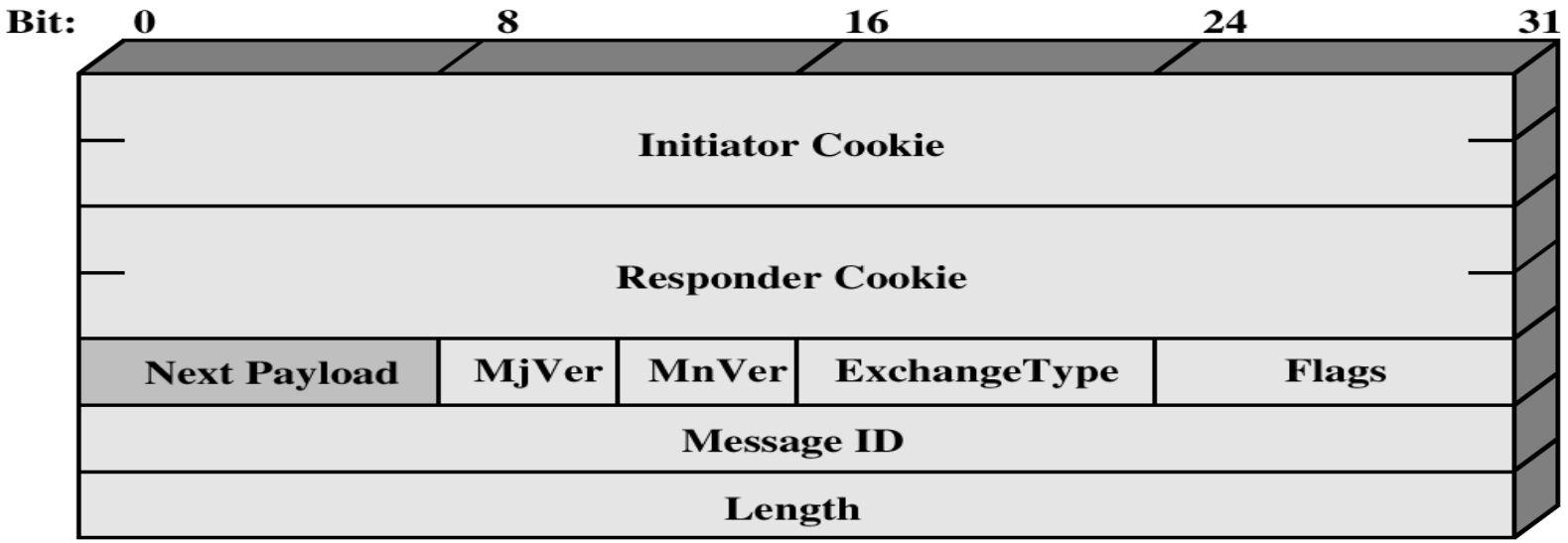
- Three authentication methods:
  - Digital signatures
    - Signed hash over information known by both
  - Public-key encryption
    - Encryption of information known by both
  - Symmetric-key encryption
    - exchanged using some out-of-band mechanism

ISAKMP

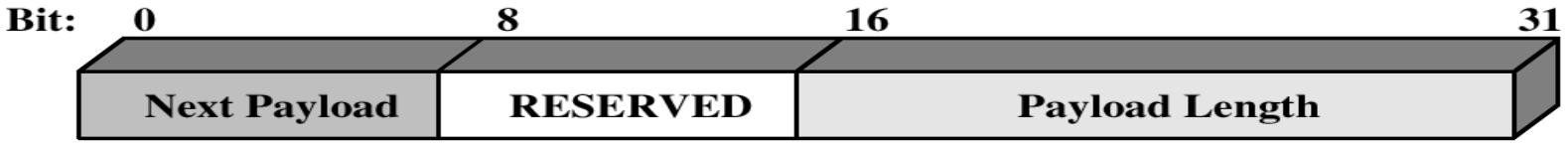
Internet Security Association and Key Management Protocol

- Provides framework for key management
- Defines procedures and packet formats to establish, negotiate, modify, & delete SAs
- Independent of key exchange protocol, encryption alg., & authentication method
- **Phase 1:** ISAKMP peers establish bi-directional secure channel using *main mode* or *aggressive mode*
- **Phase 2:** negotiation of security services for IPSec (maybe for several SAs) using *quick mode;* can have multiple Phase 2 exchanges, e.g., to change keys

ISAKMP



(a) ISAKMP Header



(b) Generic Payload Header

Figure 6.12 ISAKMP Formats

IKE Payloads & Exchanges

- have a number of ISAKMP payload types:
  - Security Association, Key Exchange, Identification, Certificate, Certificate Request, Authentication, Nonce, Notify, Delete, Vendor ID, Traffic Selector, Encrypted, Configuration, Extensible Authentication Protocol
- payload has complex hierarchical structure
- may contain multiple proposals, with multiple protocols & multiple transforms

ISAKMP Payload Types

Type	Parameters	Description
Security Association (SA)	Domain of Interpretation, Situation	Used to negotiate security attributes and indicate the DOI and Situation under which negotiation is taking place.
Proposal (P)	Proposal #, Protocol-ID, SPI Size, # of Transforms, SPI	Used during SA negotiation; indicates protocol to be used and number of transforms.
Transform (T)	Transform #, Transform-ID, SA Attributes	Used during SA negotiation; indicates transform and related SA attributes.
Key Exchange (KE)	Key Exchange Data	Supports a variety of key exchange techniques.
Identification (ID)	ID Type, ID Data	Used to exchange identification information.
Certificate (CERT)	Cert Encoding, Certificate Data	Used to transport certificates and other certificate-related information.
Certificate Request (CR)	# Cert Types, Certificate Types, # Cert Auths, Certificate Authorities	Used to request certificates; indicates the types of certificates requested and the acceptable certificate authorities.
Hash (HASH)	Hash Data	Contains data generated by a hash function.
Signature (SIG)	Signature Data	Contains data generated by a digital signa wture function.
Nonce (NONCE)	Nonce Data	Contains a nonce.
Notification (N)	DOI, Protocol-ID, SPI Size, Notify Message Type, SPI, Notification Data	Used to transmit notification data, such as an error condition.
Delete (D)	DOI, Protocol-ID, SPI Size, # of SPIs, SPI (one or more)	Indicates an SA that is no longer valid.

ISAKMP Exchange Types

Exchange	Note
(a) Base Exchange	
(1) **I → R:** SA; NONCE	Begin ISAKMP-SA negotiation
(2) **R → I:** SA; NONCE	Basic SA agreed upon
(3) **I → R:** KE; ID_I; AUTH	Key generated; Initiator identity verified by responder
(4) **R → I:** KE; ID_R; AUTH	Responder identity verified by initiator; Key generated; SA established
(b) Identity Protection Exchange	
(1) **I → R:** SA	Begin ISAKMP-SA negotiation
(2) **R → I:** SA	Basic SA agreed upon
(3) **I → R:** KE; NONCE	Key generated
(4) **R → I:** KE; NONCE	Key generated
(5)* **I → R:** ID_I; AUTH	Initiator identity verified by responder
(6)* **R → I:** ID_R; AUTH	Responder identity verified by initiator; SA established
(c) Authentication Only Exchange	
(1) **I → R:** SA; NONCE	Begin ISAKMP-SA negotiation
(2) **R → I:** SA; NONCE; ID_R; AUTH	Basic SA agreed upon; Responder identity verified by initiator
(3) **I → R:** ID_I; AUTH	Initiator identity verified by responder; SA established
(d) Aggressive Exchange	
(1) **I → R:** SA; KE; NONCE; ID_I	Begin ISAKMP-SA negotiation and key exchange
(2) **R → I:** SA; KE; NONCE; ID_R; AUTH	Initiator identity verified by responder; Key generated; Basic SA agreed upon
(3)* **I → R:** AUTH	Responder identity verified by initiator; SA established
(e) Informational Exchange	
(1)* **I → R:** N/D	Error or status notification, or deletion

Notation:
- I = initiator
- R = responder
- * = signifies payload encryption after the ISAKMP header

Encryption and Authentication Algorithms, IPsec v2

- Encryption:
  - Three-key triple DES
  - RC5
  - IDEA
  - Three-key triple IDEA
  - CAST
  - Blowfish
- Authentication:
  - HMAC-MD5-96
  - HMAC-SHA-1-96

Encryption and Authentication Algorithms, IPsec v3

- Encryption:
  - AES
- Authentication:
  - AES based CMAC (Cipher-based MAC)
- Additionally there are cipher suits defined for NSA with even higher security

AES-CMAC

- AES-CMAC achieves a security goal similar to that of HMAC.
- Since AES-CMAC is based on a symmetric key block cipher, AES, and HMAC is based on a hash function, such as SHA-1, AES-CMAC is appropriate for information systems in which AES is more readily available than a hash function (RFC 4493)

Some Limitations of IPsec

- IPsec cannot provide end-to-end security as systems work at higher levels
  - e.g.: if you need emails encrypted from the sender's desktop and decrypt them at the receiver's site)
- Specific applications have particular requirements on security and IPsec does not provide all security services:
  - e.g.: IPsec cannot provide total security for credit card payment systems

Alleged NSA interference

- There have been several allegations against NSA to put in backdoors in IPsec or making limitations in the strength of the cryptos used making it possible to easily brute force key exchanges.
- This is quite possible but never proven, but it can be a reason to avoid IPsec

Source: https://en.wikipedia.org/wiki/IPsec

Virtual Private Network (VPN)

- A VPN is one or more secure connections over an unsecure public network
- You can implement a VPN using several different protocols e.g.
  - IPsec (IP Security typically in tunnel mode)
  - PPTP (Point-to-Point Tunnelling Protocol)
  - SSL/TLS (Secure Socket Layer/Transport Layer Security)
  - SSH (Secure shell)
  - L2TP (Layer 2 Tunnelling Protocol)
  - OpenVPN

New strong contenders for VPN Tailscale/Nebula and WireGuard

- ”Btw, on an unrelated issue: I see that Jason actually made the pull request to have WireGuard included in the kernel. Can I just once again state my love for it and hope it gets merged soon? Maybe the code isn't perfect, but I've skimmed it, and compared to the horrors that are OpenVPN and IPsec, it's a work of art.” (Linus Torvalds, 2018)
- WireGuard is a communication protocol and free and open-source software that implements encrypted virtual private networks (VPNs) and was designed with the goals of ease of use, high speed performance, and low attack surface. It aims for better performance and more power-saving than the IPsec and OpenVPN tunnelling protocols. The WireGuard protocol passes traffic over UDP.
- Tailscale is built on top of WireGuard. To connect devices using Tailscale, you install and log in to Tailscale on each device. Tailscale manages key distribution and all configurations for you. This can be particularly useful if some of the devices belong to non-technical users.
- A option to Tailscale is Nebula (built by the same team that created Slack)

Sources: https://www.grc.com/sn/sn-830-notes.pdf
https://tailscale.com/kb/1148/tailscale-vs-nebula/